Similar Language Translation

June 25, 2019

Degree's Thesis

Telecommunications Technologies and Services Engineering

Autor: Lluís Guàrdia Fernàndez

Director: Marta R. Costa-Jussà

Abstract

Similar Languages are an interesting line of research within Machine Translation since it settles the perfect scenario to exploit the commonalities that present these similar languages. This contrasts other Machine Translation tasks on languages that are more distant and can not exploit such similarities. In this project, we work with the similar languages pairs of Czech-Polish and Spanish-Portuguese.

In this work, we are comparing two of the most popular approaches in automatic translation: statistical and neural-based systems. The latter is the current approach that is used by important companies like Google.

During the project execution, we successfully participated in the 1st WMT Similar Language Translation Task with the submission of the TALP-UPC system using both statistical and neural systems, which was placed 1st for Czech-Polish and 2nd for Spanish-Portuguese in the official evaluation.

To improve the results obtained, it is proposed and analyzed the use of a combination of both systems mentioned with back-translation as a metric measure.

Obtaining in the Spanish-Portuguese case a result 6 BLEU points greater with statistic model than neural, while in Czech-Polish the neural outperforms by 2 BLEU points the statistical. Between both systems, there is a difference of about 40 BLEU points in quality. With the obtained results it is concluded that both analyzed systems achieve very similar results which performances depend on the language pair analyzed.

Also is inferred that our proposed system combination doesn't contribute with any substantial improvement, actually sometimes it could worsen the obtained results. It is due to back-translation not being able to be considered a good metric to evaluate a translation system knowing, among other reasons, the low correlation values between the quality of the obtained translation and the quality of its back-translation.

Resum

La traducció de llengües similars és una secció en la traducció automàtica que sempre ha generat interès en recerca per tal de buscar la forma d'aprofitar la similitud que presenten aquestes llengües envers altres de més llunyanes gramaticalment. En aquest projecte es treballa amb els parells de llenguatges similars Xec-Polac i Espanyol-Portuguès.

En aquest treball comparem dos dels enfocaments més populars en traducció automàtica: els sistemes estadístic i neuronal. El segon és l'enfocament utilitzat actualment per grans companyies com Google.

Durant l'execució del projecte també es participa en la 1a Tasca de Traducció de Llengües Similars en WMT amb la submissió del sistema TALP-UPC utilitzant ambdós sistemes, tant estadístic com neuronal, obtenint un 1r lloc en Xec-Polac i un 2n lloc en Espanyol-Portuguès en la avaluació oficial.

Per tal de millorar els resultats obtinguts, es proposa i s'analitza l'ús de la combinació dels dos sistemes mencionats utilitzant back-traducció com a mètrica de mesura.

Obtenint-se en el cas Espanyol-Portuguès un resultat 6 punts BLEU major amb el model estadístic que amb el neuronal, mentre en Xec-Polac el neuronal supera per 2 punts BLEU l'estadístic. Entre els dos sistemes s'obté una diferència d'aproximadament 40 punts BLEU en qualitat. Amb els resultats obtinguts es conclueix que els dos sistemes analitzats obtenen resultats molt similars sent el seu rendiment dependent en gran mesura del parell de llengües analitzades.

També s'infereix que la utilització de la combinació de sistemes no aporta cap millora substancial, de fet pot arribar a empitjorar, als resultats obtinguts, degut a que no es pot considerar la back-traducció com a una bona mètrica per avaluar un sistema de traducció sabent, entre altres raons, la mala correlació entre la qualitat de la traducció obtinguda i la qualitat de la seva back-traducció.

Resumen

La traducción de lenguas similares es una sección en la traducción automática que siempre ha generado interés en investigación para buscar la forma de aprovechar la similitud que presentan estas lenguas delante otras más lejanas gramaticalmente. En este proyecto se trabaja con los pares de lenguajes similares Xeco-Polaco y Español-Portugues.

En aquest treball comparem dos dels enfocaments més populars en traducció automàtica: els sistemes estadístic i neuronal. El segon és l'enfocament utilitzat actualment per grans companyies com Google.

En este trabajo comparamos dos de los enfoques mas populares en traducción automática: los sistemas estadístico y neuronal. El segundo es el enfoque usado actualmente por grandes compañias como Google.

Durante la ejecución del proyecto también se participa en la 1a Tasca de Traducción de Lenguas Similares en WMT con la sumisión del sistema TALP-UPC usando ambos sistemas, tanto el estadístico como el neuronal, obteniendo un 1r puesto en Xeco-Polaco y un 2o puesto en Español-Portugues en la evaluación oficial.

Para mejorar los resultados obtenidos, se propone y se analiza el uso de una combinación de los dos sistemas mencionados usando back-traducción como métrica de mesura.

Obteniéndose en el caso Español-Portugues un resultado 6 puntos BLEU mayor con el modelo estadístico que con el neuronal, mientras en Xeco-Polaco el neuronal supera por 2 puntos BLEU el estadístico. Entre los dos sistemas se obtiene una diferencia de aproximadamente 40 puntos BLEU en calidad. Con los resultados obtenidos se concluye que los dos sistemas analizados obtienen un resultado muy similar siendo su desempeño dependiente en gran mesura de el par de lenguajes analizados.

También se infiere que la utilización de la combinación de sistemas no aporta ninguna mejora substancial, de hecho puede llegar a empeorar, a los resultados obtenidos, debido a que no se puede considerar la back-traducción como una buena métrica para evaluar un sistema de traducción sabiendo, entre otras razones, la mala correlación entre la calidad de la traducción obtenida y la calidad de su back-traducción.

Acknowledgments

First I want to thank my tutor Marta R. Costa-Jussà for guiding me through this project and giving me the opportunity to participate in a task presented internationally, and Magdalena Biesialska. who was part of the project too and without her would have been impossible to finish it. It was a pleasure to work with you.

I want to thank my parents for all the support, dedication, affection and all the invaluable things they gave me during all my life.

Thanks to all my colleagues who worked with me during this long five years either in the laboratory or theoretical classes. All names couldn't fit in this page but I want to thank especially Xavier Barrera and Oriol Barbany, who had been in almost every laboratory class during this last two years.

And finally, I want to thank Maria, who has infinite patience for bearing me and listening all my verbosity for the lasts months, and was at my side whenever I needed it.

Revision history and approval record

Revison	Date	Purpose
0	20/05/2019	Document creation
1	21/06/2019	Document revision
2	24/06/2019	Document approbation

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Lluís Guàrdia Fernàndez	lluis.guardia.f@gmail.com
Marta R. Costa-Jussà	marta.ruiz@upc.edu
Magdalena Biesialska	magdalena.biesialska@upc.edu

Written by:		Reviewed and approved by:	
Date	20/05/2019	**Date**	24/06/2019
Name	Lluís Guàrdia Fernàndez	**Name**	Marta R. Costa-Jussà
Position	Project Author	**Position**	Project Supervisor

Contents

List of Figures

List of Tables

1 Introduction

From long to the past, humans are interested in the possibility of communicating in an automatic way between different languages, from Johan Joachim Becher on 1661, with the first MT resembled approach system [Becher1962], until the more recent Alan Turin, and his decipherment of the German Enigma machine, during WW2 [Lee1997].

Since a few years ago, especially thanks to the progress on Machine Learning, the increased number of source texts available in more and more languages from the internet and the improvement on accessibility, both for companies and consumers; the Machine Translation (MT) systems have been improving a lot, and they still do it today, having the translation of documents without the help from any other person as their goal.

In the last years, the similar language translation may have started to wrongly been considered as a solved task, which, as the name indicates, consist of translating languages that are more close due to their similar point of origin, and are more easy to translate (like the Spanish and Portuguese). That is mostly due to the great results obtained by the MT systems.

These MT systems are automatic translation systems or "translation carried out by a computer", as defined in the Oxford English dictionary. In a very summarized form: it's a process sometimes referred to as Natural Language Processing [Weiscbedel et al.] where you input a text in a certain language to the computer and, in result, it gives you the text translated to the target language.

However, there are still some challenges to surpass that will lead to a better system performance in the future, such as limited resources to some of the less known languages, out-of-domain, or the difference between alphabets used in both languages, even at similar languages, as you could observe at Table 1[1] [2].

Within this systems, the most distinguished for its results is the Neural MT [Vaswani et al.2017], which uses Deep Learning to generate, using the source texts, information vectors for each word associating information of the words surrounding it, capturing a great amount of information

Despite this, for some of the tasks, statistical approaches are still competitive [Lample et al.2018].

[1] There are two versions of the Hungarian alphabet, one which is said to be official (also characterized as 'full' or 'old'). The other one is said to be taught in school and is characterized as 'strict' or 'standard'. In this case we are referring to the full version

[2] The alphabet doesn't include the letters with diacritics where the resulting letter is considered an ordinary letter in the alphabet of the language where it is used

Alphabet	Langauge
26	Standart European

21	Italian	
23	Portuguese	
26	English German Irish	French Icelandic
27	Finnish	
29	Danish Norwegian Turkish	Farese Swedish Sami
30	Spanish	Croatian
31	Czech	Romanian
32	Estonian Polish	Lithuanian
33	Latvian	
36	Albanian	Bosnian
44	Hungarian	

Table 1: Alphabet size in European languages [Leira]

1.1 Statement of purpose and contributions

The main goal of the project is to test which system among neural or statistical MT is better for close languages with limited results, implementing these statistical and neural MT, and participate in the 1st Similar Language Translation WMT task for the Czech-to-Polish and Spanish-to-Portuguese translation directions. The main contribution is the implementation of a statistical MT with Moses [Koehn et al.2007] system which ranked second in the Similar Language Translation Shared Task in the Fourth Conference on Machine Translation (WMT19). Also the usage of the techniques of back-translation and minimum Bayes risk [Kumar and Byrne2004] in order to evaluate the translations.

1.2 Requirements and specifications

This project has been developed in two differentiated parts, that were combined in the last sections. For the Neural MT was used the open-source Fairseq architecture [3] which required PyTorch version greater or equal than 1.0.0 and Python version greater or equal than 3.6. And for the statistical MT, we used the open-source Moses toolkit v4.0. [4]

All the software has been launched in the CALCULA cluster, which consist of 8 servers from the TSC department of the UPC, each with 2 IntelR©XeonR©E5-2670 v3 2,3GHz 12N processors,

[3] https://github.com/pytorch/fairseq
[4] https://github.com/moses-smt/mosesdecoder

and a total of 16 NVIDIAGTX Titan X GPUs. Each GPU has 12GB of memory and 3072 CUDA Cores. Among those servers, there was a great heterogenous variety of CPU's for this task due to the difference of ages of them. The specifications for them are in the table 2:

	processor	model name	cpu MHz	cache size	cpu cores
veuc01	39	Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz	2399.864	25600KB	10
veuc05	47	Intel(R) Xeon(R) CPU E5-2670 v3 @ 2.30GHz	1915.323	30720 KB	12
veuc06			2599.866		
veuc07	23	Intel(R) Xeon(R) CPU X5660 @ 2.80GHz	1652.740	12288 KB	6
veuc08			1606.261		

Table 2: Specifications of the cpu's in CALCULA machines

1.3 Methods and procedures

This project main idea was originally proposed by my supervisor and it didn't come from any previous work. It is a combined effort between me and Magdalena Biesialska, who carried out the Neural MT tasks.

In this project, the neural model is based on the Transformer architecture implemented by Facebook in the Fairseq toolkit. The transformer is the most current state-of-the-art NMT architecture [Vaswani et al.2017] which relies solely on the self-attention mechanism and shows significant performance improvements over traditional sequence-to-sequence models.

The statistical model is based on a Phrase-based structure implemented by the open-source Moses toolkit, which is one of the most widely used Statistical MT Application.

The majority of the coding during this project was written in Bash scripting.

1.4 Work Plan

The project was structured in the Work Packages exposed below and the Gantt Diagram.

- WP 1: Project propose and work plan
- WP 2: Information research
- WP 3: Preparing the SMT model (Moses)
  - · data preparation
  - · building the system
- WP 4: Translation for NMT model [5]

[5] Section 6.2

- WP 5: WMT Paper [6]
- WP 6: Critical review
- WP 7: Evaluation
- WP 8: Final Report
- WP 9: TFG presentation

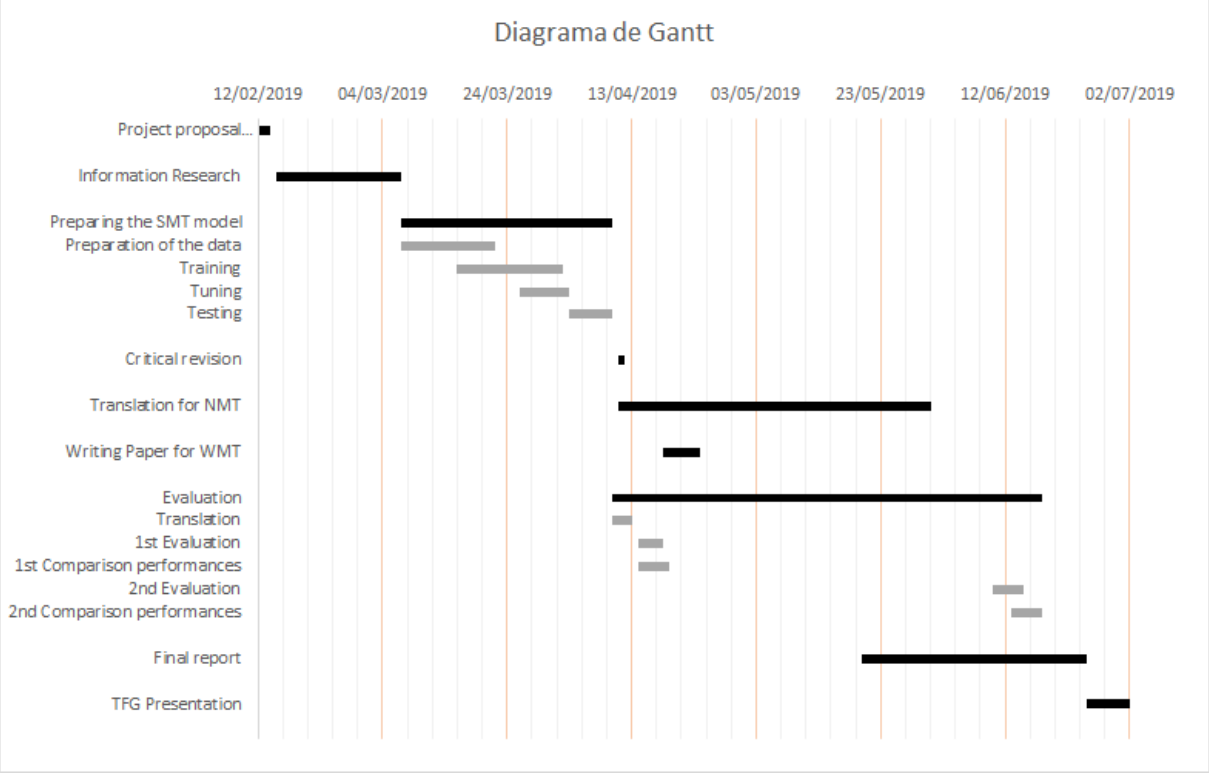

Figure 1: Gantt Diagram

1.5 Incidences

In order to use as a pseudo-corpus for the Neural MT we needed to translate the corpus using the Statistical MT, this process took more memory and time that we had expected. As it was a huge corpus, it also took more time to train the Neural MT system. Due to this, for the Spanish-Portuguese language pair, we were unable to finish training our NMT model with the pseudo corpus (Table 7).
In the SMT case, the translation rate was normally approximately 2000 sentence/hour, but it undulates between 500 and 3.000 depending on which server was allocated within the CALCULA cluster.

[6] Section 2

2 WMT Task

In this section we explain the shared task in which we participated.

The shared task: Similar Language Translation [7] is proposed for the 4th Conference on Machine Translation (WMT), which will be held on August 1-2, 2019, in Florence, Italy. It is organized by Costa-jussà, M. from Universitat Politècnica de Catalunya, Malmasi, S. from Harvard Medical School, Pal, S. from Saarland University, and Zampieri, M. from University of Wolverhampton.

WMT started in 2005 and, since then, a wide range of shared task related to translation has been carried out. From translation of news, up to unsupervised translation, through translation using system combination and others.

A shared task consists of a challenge provided by the organizers with the training data attached, it has to be remarked that it is a combined effort to improve and research in the field, not a competition between the participants. Then, on a pre-announced data, unseen test sets are released for the use of participants, which will have to submit the system results given these sets. Finally, the results are published and the evaluation and techniques are presented in a conference.
For the first time in WMT, a shared task in "Similar Language Translation" is organized. The objective of this task is to evaluate the performance of the translation between pairs of similar languages from the same language family, using state-of-the-art translation systems.

In the increased use of MT technologies, there is also an increased interest in training the systems between languages different than English, since in automatic translation it is, by far, the most used language, due to the great quantity of training data and resources available. The vast majority of MT systems look for translation from/to English or, in case of translation of languages with a shortage of resources available, it is used as the pivot language.
The fact that English is not used in any of the languages for/to translate supposes a lesser quantity of parallel data available for the task. In order to overcome this limitation, it's necessary to find a way to take advantage of the similarity between the pair of languages in a direct translation of similar languages.

The evaluation in this task will be carried out using automatic evaluation metrics. All systems are ranked by BLEU score [Papineni et al.2002], and TER score [Snover et al.2006] will be calculated for systems with BLEU scores greater than 5.0.

In this task are only allowed submissions which only uses the parallel data provided (constrained) for training, no additional parallel data is allowed. With monolingual data, you are encouraged to develop novel solutions to improve translation quality.
All participants will be provided with training and testing data for 3 pairs of languages: [8]

- Spanish - Portuguese (Romance languages)
- Czech - Polish (Slavic languages)
- Hindi - Nepali (Indo-Aryan languages)

[7] http://www.statmt.org/wmt19/similar.html
[8] in this project only the two first will be used

3 Background

In this section, we overview the statistical (phrase-based) and the neural-based MT approaches that we use in this study, and the evaluation metric used, the BLEU score.

3.1 Statistical Machine Translation

Statistical MT translations are based in the statistic model and information theory, it could be that the probability of a word combination in the source language could be calculated from the statistic relationships between this words, extrapolating these relationships from the aligned corpus [9] from bilingual texts (a famous example of this kind of bilingual text is the Rosetta Stone), as Jordan, Dorr and Benoit express [Dorr et al.1998]. This approach contrasts with previous traditional approaches like the automatic translation based on rules, which were used to express translation knowledge, or based on examples.

Although the SMT approach is considered to be first proposed in 1990 by Peter F. Brown [Brown et al.1990] and being a hot topic since then, the first to suggest the application of the ideas of statistics models and information theory to the machine translation field was W. Weaver at 1949 [Weaver1995], but it didn't draw the attention until it was reintroduced by Brown and the IBM researchers.

The researchers from the T.J. Watson research center with IBM were the first to propose an SMT based on source channel model [Brown et al.1993], also called noisy channel model (Figure 2). This process is divided into three sub-problems: the modelling of a language model, the modelling of a translation model and decoding.

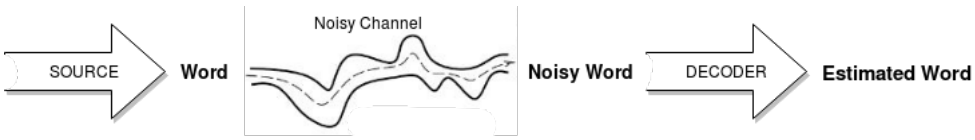

Figure 2: Noisy channel concept

Using the Bayes rule to reformulate the translation probability for translating a source sentence s into the target language t as:

$$argmax_t P(t|s) = argmax_t P(t)P(s|t)$$

$P(t)$ is the language model, which takes care of the fluency in the target language. It is obtained through monolingual corpora in the target language. Basically, it estimates how probable a sentence is, but it has problems such as zero probability in long chains, since it is difficult to observe them in the corpora. The solution to this problem is using the n-gram approach [Kneser and Ney1995]. It consists in considering the probability of a sentence as the product of the conditional probabilities of each word. For example, using a 3-gram model:

[9] For any help with the glossary: http://www.statmt.org/moses/glossary/SMT_glossary.html

The girl was upset.

$$P(t) = P(The|\phi,\phi) * P(girl|\phi,The) * P(was|The,girl) * P(upset|girl,was) \quad (1)$$

However, with this approach long-range dependencies are lost, and some n-grams can be not observed in the corpora, so smoothing techniques such as linear interpolation or back-off models are required.

Then, $P(s|t)$ is the Translation model, which is an estimation of the lexical correspondence between languages and it's obtained through the aligned bilingual corpora in both, source and target languages To generate this model, it should take into account for each word in the source language its translation, the number of necessary words in the target language, the position of the translation within the sentence and the number of words that need to be generated from scratch. In conclusion, its quality depends on the obtained word alignment, and we can estimate it with the statistical model (counting probabilities in a huge corpus) but the corpus is not aligned word by word. In this case, we could estimate word alignments together with the parameters used [Och et al.1995] or we could apply the phrase-based approach.

Finally, the $argmax_t$ part is done by the decoder. Once we have the given models (Language Model, Translation Model or others), the decoders are responsible for constructing the possible translations and searching the most probable one. There are some possibilities for this search, the most efficient and used one being beam search [Koehn et al.2003] along with cube pruning [Chiang2007].

This heuristic consists in store the B top possible word translations, where B is a threshold parameter. Then repeating this process but only to the stored options. This approach allows saving a lot of memory by not going down all the possible translations. You can see a scheme of it in figure 3.

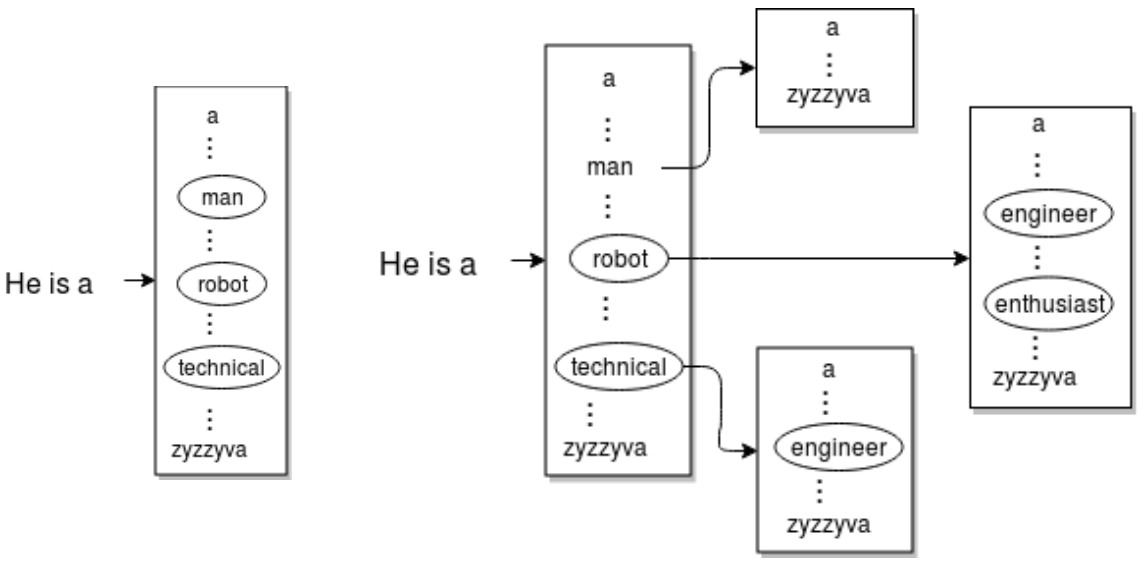

Figure 3: Basic concept of Beam Search

This source channel method was the most used and studied for automatic translation, until 2016, where it began to be replaced by the neuronal MT.

From the linguistic knowledge perspective, the framework can be classified into three models: word-based, phrase-based and syntax-based. The most usual, being used by companies like Google, IBM, ISI and others; and the one that we will focus on in this project is the phrase-based.

3.1.1 Phrase-based approach

Phrase-based statistical MT [Koehn et al.2003] uses the noisy channel model but, unlike the word-based approach which translates word by word, translates by concatenating at a phrase level the most probable target given the source text.

Source	The enemy team gave up
Word-based	El equipo enemigo paso arriba
Phrase-based	El equipo enemigo abandono

Table 3: Results comparison between word and phrase based

In this context, a phrase is a sequence of words, ignoring if it's a phrase or not from a linguistic point of view. Phrases are extracted based on the probabilistic study of a large parallel corpus, which identifies and ranks each phrase with several features, such as conditional probabilities. The collection of scored phrases constitutes the translation model.

This approach seems like a closer take to the syntax of the languages, and so allows improve on the word-to-word translation, and the phrase-learning helps to resolve ambiguities, as context can provide useful clues about translation.

In order to calibrate the output size in this approach, a W factor is introduced, which corresponds to the word cost, for each generated word in the target language [Koehn et al.2003]. So the formula remains as:

$$argmax_t P(t|s) = argmax_t P(t)P(s|t)w^{length(t)}$$

In addition to the models mentioned previously, there are also other models to help achieve a better translation, such as the reordering model, which helps in a better ordering of the phrases. The weights of each of the models are optimized by tuning over a validation set. Based on these optimized combinations, the decoder uses beam search to find the most probable output given an input. Figure 4 shows a diagram of the phrase-based MT approach.

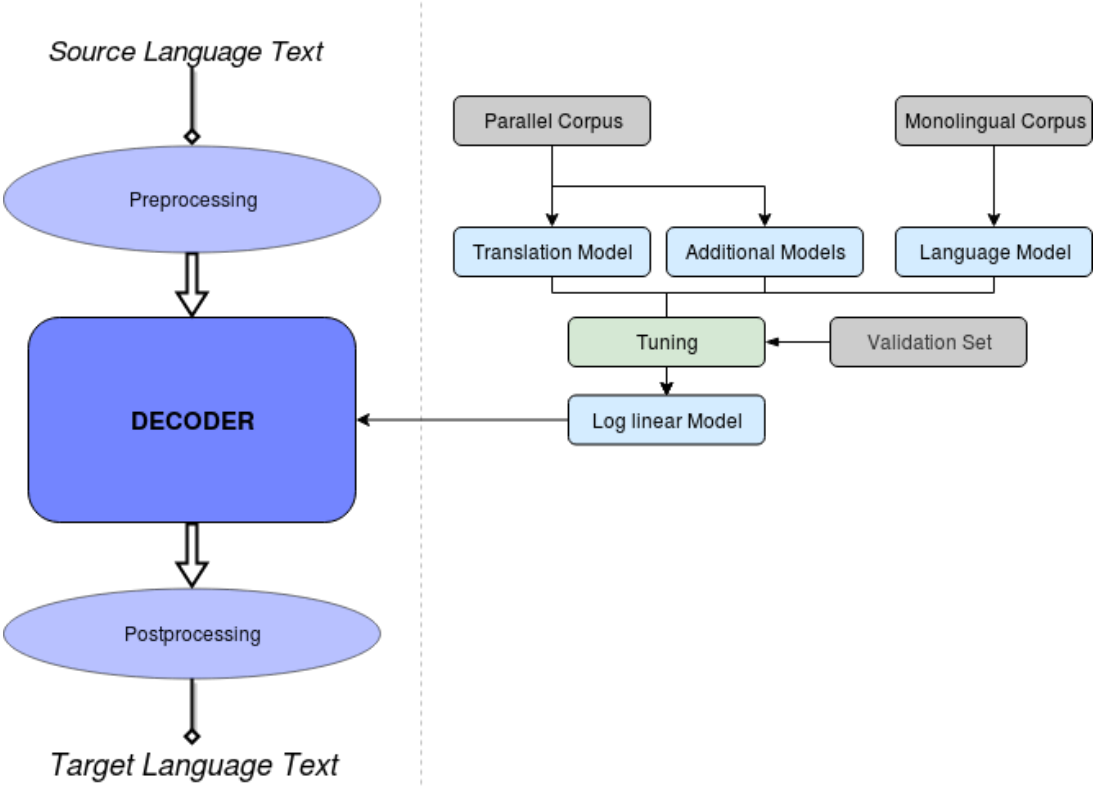

Figure 4: Basic schema of a Phrase-based MT system

3.2 Neural Approach [10]

Neural Networks for MT scientific papers started appearing around 2014 [Bahdanau et al.2014], and are very successful since then, helped by a great number of advances in recent years. The first appearance of NMT systems in an MT public competition was in 2015 (OpentMT'15). And at the OpenMT'16 the following year, 90% of the winners were NMT systems [Bojar et al.2016] showing a similar or even better performance than the phrase-based SMT systems [Kalchbrenner and Blunsom2013, Cho et al.2014, Sutskever et al.2014, Bahdanau et al.2014, Sennrich et al.2016a, Zhou et al.2016, Wu et al.2016].

NMT derives from SMT phrase-based approaches [Wołk and Marasek2015] and uses large artificial neural networks, its great difference is the use of vector representations or embeddings for words and internal states. The structure is more simple than phrase-based models since there is no separation between the Language Model, Translation Model and Reordering Model, just one sequence model that predicts one word at a time. However, this sequence predictor is conditioned by the entire source sequence and the already translated part.

Actually, the most predominant NMT model used is the bidirectional RNN provided with a Long-Short Term Memory (LSTM) units [Hochreiter1997] or Gated Recurrent Units (GRU) [Cho et al.2014] in both encoder, which is used to encode the source sentence, and decoder, which

[10] Even though I didn't participate in this part, I feel necessary to explain it since we did use its translations results

does the word prediction in the target language [Bahdanau et al.2014]; combined with an attention mechanism [Luong et al.2015]. Other approaches, although less usual, are used for sequence modelling such as Convolutional Neural Networks (CNN) [Kalchbrenner et al.2016, Gehring et al.2017].

In this project we will focus on one of the more current architectures, the Transformer [Vaswani et al.2017], which shows an important improvement over sequence-to-sequence traditional models. In spite of using some earlier concepts used in RNN-CNN based models such as residual connections [He et al.2015] or position embeddings [Gehring et al.2017].

Since 2016, the majority of the best MT are using Neural Networks [Bojar et al.2016] such as Google, Microsoft, Yandex, among other translation services. An open source neural machine translation system, OpenNMT, has been released by the Harvard NLP group [Klein et al.2018].

3.2.1 Artificial Neural Networks

Artificial neural networks (ANN) are a type of machine learning algorithm inspired by the functioning of biological neurons in animals brains. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules.
They normally consist of a group of basic processing units called artificial neurons (AN) or perceptrons (Figure 5) which are wired together in a complex communication network. They can compute an output given input data by decomposing it in different representations in order to identify different characteristics.

Each AN model is a simplified model of a real neuron, which sends off a new signal if it receives strong enough input signal from the other nodes to which is connected, allowing it to perform some basic operations such as AND, OR or NOR.
This model was first proposed by Warren McCulloch and Walter Pitts in 1943 [McCulloch and Pitts1943]. Among many other proposed models, the most simple AN architecture was the perceptron [Rosenblatt1961], which improves the usage of binary values for the McCulloch and Pitts model to being operational with any numbers. It works through an algorithm that computes the so-named activation function:

$$ouput = f(\sum_{\forall i} w_i x_i + b) \tag{2}$$

Where w_i are the weights of the input values x_i and b is a bias, used to give some extra degree of freedom. These values are computed using gradient descent techniques [Barzilai and Borwein1988], which consist in taking proportional steps to the negative of the gradient of the function iteratively at the current point, so it approaches the global minimum of the function.

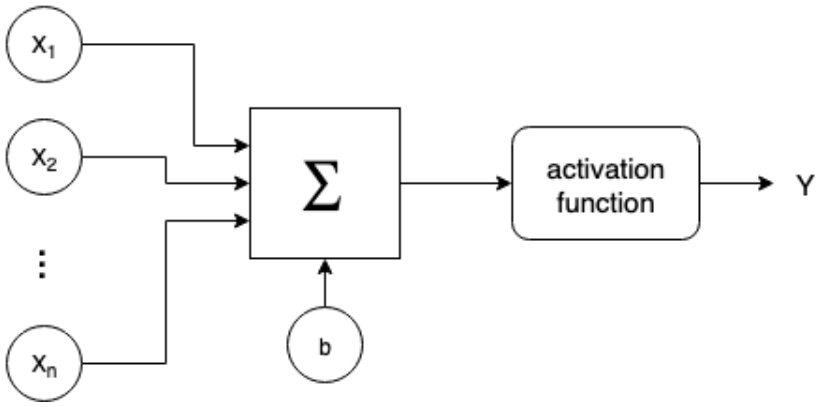

Figure 5: Structure of a perceptron

The output (Y in Figure 5) has an internal threshold, so the output values of the perceptron are binary and they depend on if the output of the activation function exceeds or not this threshold. This allows the perceptron to linearly separate samples into two classes, that's why it can compute basic operations like AND or OR, but not non-linear separable functions or problems like XOR (Figure 6).

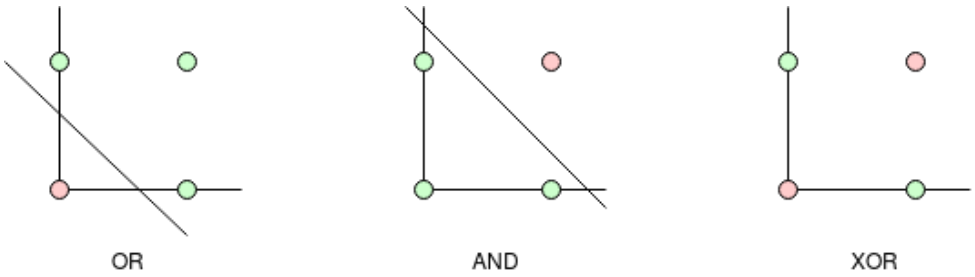

Figure 6: Operations AND, OR and XOR as linear separation problems

To solve this more complex structures are needed. One basic neural network structure, among many others, is the Multilayer Perceptron (MLP), which basically consist of multiple layers of perceptrons.
The basic structure consists of 3 layers of nodes: the hidden layers for multiple representations of data and characteristic identification, these layers are fed with all the data by an input layer, and finally the output layer, which can use a different activation function depending on the nature of the task.

Adding Monolingual Data. Differently, from the statistical MT approach, the neural MT approach does not include monolingual data in the standard training. However, previous studies have reported notable improvements by adding monolingual corpora through back-translation [Sennrich et al.2016b].

3.2.2 Recurrent Neural Network

Recurrent Neural Network (RNN) is a type of ANN formed by a sequence of concatenations of the same unit along a temporal sequence (Figure 7), this structure allows them to retain information from previous data like a temporal memory. This approach is used in the majority of NMT models today and were based on David Rumelhart's work in 1986 [Williams et al.1988].

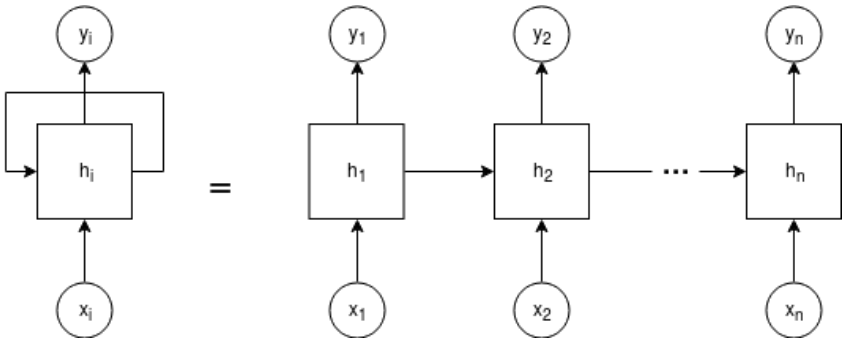

Figure 7: Diagram of an RNN

Due to its capacity to retain information, they are pretty useful for sequential data, where each element of the sequence could be related to others.
Another type of RNN with improved long term dependencies are the LSTM [Hochreiter1997], which, due to its internal structure formed by four operation layers unlike the just one used by conventional RNN, can perform different operations such as update, forget or output information. A variant of the LSTM are the Gated Recurrent Unit (GRU) [Cho et al.2014], which have a simpler structure, making them computationally more efficient and faster to train.
These are the more common RNN in NMT, however, all of them suffers from the vanishing gradient problem, which means that they are losing more information as "time" passes.

3.2.3 Transformer

One architecture proposed that don't suffer from memory loss is the Transformer [Vaswani et al.2017], which relies only on the attention mechanism without resorting to recurrence nor convolution [Luong et al.2015].
The attention mechanism looks at the input sequence and decides at each step which other parts of the sequence are important by giving them different weights (Figure 8) that will be used by the decoder as shown in Figure 4 [11].

Summed up, each word receives an attention weight normalized between 0 and 1, which is defined by how each word of the sentence is influenced by all the other words in the sequence.

[11] This example weights had not been corroborated to be exact, is more a concept example

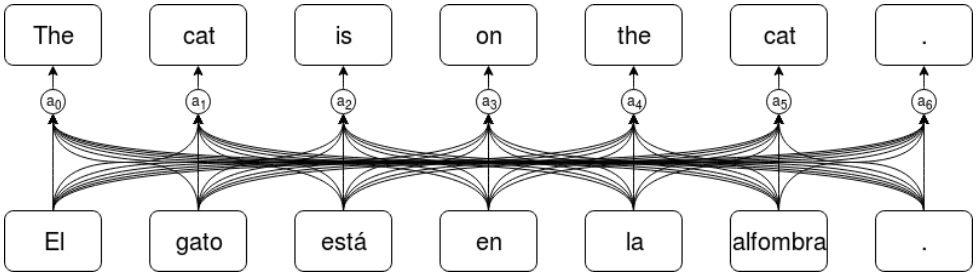

Figure 8: Attention mechanism concept

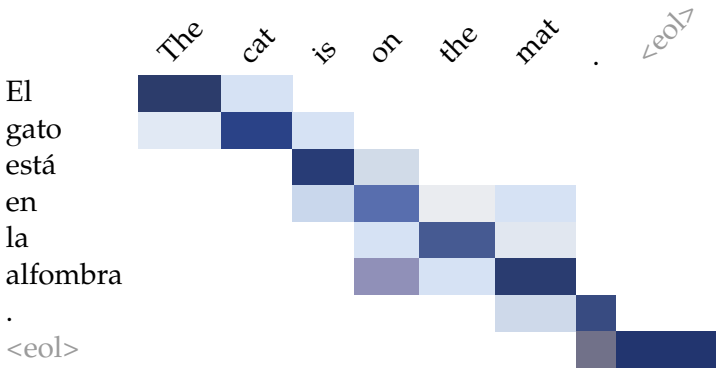

Table 4: Wheights in an attention mechanism example

In the variant used in this project of the implemented self-attention by the Transformer, the representation of a word given is produced by means of computing a weighted average of attention scores for all the words of a sentence.

3.3 BLEU score

Imagine you have a Spanish sentence, and you are given a human-generated translation of it as a reference. However, there could be multiple sentences considered perfectly good translations of that Spanish sentence.

The Bilingual Evaluation Understudy or BLEU [Papineni et al.2002] is a method to evaluate the quality of a machine-translated text. The basic idea behind BLEU is that, the closer a machine translation is to a professional human, the better it is. BLEU allows using more than one reference, which allows better robustness.

In order to illustrate better how the BLEU score works we will use an example, with the Spanish sentence ***El gato está en la alfombra*** as the sentence to translate. As a human reference, we have gotten two accepted translations, the first being ***The cat is on the mat***, and the second being ***There is a cat on the mat***.

What BLEU method does is, given a machine-translated text, it computes a BLEU score that measures how good that MT is. The more basic intuition behind the BLEU score is to look at the machine-generated output and see if the words it generates appear in the human-generated

references. So, if we look at each word in the MT output and see if it appears in the references, we are calculating the precision of the MT output, which is a number between 0 and 1.

However, in order to resolve some deficiencies, it's normally used the modified precision measure in which we will give each word credit only up to the maximum number of times it appears in the reference sentences. Also, as we don't want to just look at isolated words, we will look at n-grams (a group of n consecutive words).

And so the algorithm goes as follows:

- First, we will count the number of distinct n-grams in the candidate.
- Then we will count the number of times each n-gram g_i occurs in each reference. But we will only take the maximum of each of these values calculated.
- Finally, we will add the maximum calculated previously of each n-gram, and divide them by the total number of n-grams in the candidate (this time they don't have to be distinct).

So to start with the example, we get the slightly good translation ***The cat the cat on the mat*** as an MT output, and we will evaluate it using bi-grams.

Candidate: The cat the cat on the mat
Reference1: The cat is on the mat
Reference2: There is a cat on the mat

bigrams	max count	sum of max counts	appearences	total of bi-grams	score
the cat	1	4	2	6	4/6=2/3=0.66
cat the	0		1		
cat on	1		1		
on the	1		1		
the mat	1		1		

Table 5: Results obtained for the example candidate

So, in this example, we get that the sum of each bi-gram maximum number of appearances in the references is 4. And, although we have 5 distinct bi-grams in the candidate, the number of total bi-grams is 6 as ***the cat*** appears twice. In result, we obtain a score of 4/6 or 2/3.

So we could compact all of this in the formula 3 where the subscript n indicates for what number of n-grams are we calculating and y is the MT output or candidate. :

$$p_n = \frac{\sum_{n-grams \in y} \text{max count of appearances in reference}}{\sum_{n-grams \in y} \text{total n-grams in candidate}} \tag{3}$$

Finally, to obtain the final BLEU score, we calculate the Combined BLEU score (Formula 4) which is the value of all the n-grams modified precision. Where we basically exponentiate e by the mean of values from unigrams to 4-grams and multiply it by BP, which stands for brevity penalty (Equation 5) and is an adjustment factor that penalizes translation systems that output translations that are too short.

$$score = BP * exp(\frac{1}{4}\sum_{n=1}^{4} p_n) \tag{4}$$

$$BP = \begin{cases} 1 & \text{if } length_y \leq length_{reference} \\ exp(1 - \frac{length_y}{length_{reference}}) & \text{otherwise .} \end{cases} \tag{5}$$

BLEU score was revolutionary for machine translation because it gave a, by no means perfect, but pretty good single real number evaluation metric.

In practice, BLEU was one of the first metrics to claim a high correlation with human judgements of quality [Coughlin2003] and remains one of the most popular automated and inexpensive metrics. And so there are multiple open source implementations that you can download and use to evaluate your own system (Moses multi-bleu.perl script, NIST mteval-vXX.pl script, etc), but it's recommended to stick with only one per project since they have different implementations and their results may differ between them.

4 Related Work

In this section it is explained a little overview of previous works related to the project.

A considerable quantity of works have been developed lately associated with similar languages, due to the less complexity of them against non-similar languages. Even so, a great part of them are more related to the task of distinguishing between them more than translation.

West Slavic languages, which is the case for Czech and Polish, had never been a great focus of interest in research partly due to the shortage of resources available. Lately, several systems have been implemented more focused in the translation of a third, more international, language such as English [Kirschner1987, Wolk and Marasek2014] or Russian [Bémová and Kubon1990] than in the translation between them.
In the Czech-Polish case, we find that there is a minimum of translation systems. The majority of them are about Czech-Slovak since the great similarity between both of them. Even so, some works focused on Czech-Slovak were searching for a multilingual implementation with the rest of languages within the same family, such is the case of the Kubon, V. work [Hajič et al.2000], in which they were searching for an implementation using more simple methods, following a word-for-word approach.

In the case of the romance languages, Spanish and Portuguese within them, we can find more cases of direct translation systems between similar languages, but mostly in Spanish Catalan, due to its great translation results [Alonso2005].
One approach in Catalan-Spanish taken by interNOSTRUM [Canals et al.2019] and Sishitra [Navarro Cerdán et al.2004], along with other few papers in other pair of languages such as one from Irish to Gaelic Scottish [Scannell2006], were more focused towards exploring similarities in varieties, dialects and closely related languages consisting of a pipeline of different components such as a Part-of-Speech tagger (POS-tagger) or a Naïve Bayes word sense disambiguator. In the more specific case of Spanish-Portuguese, Garrido-Alenda used a word-for-word MT refined by shallow parsing techniques [Garrido-Alenda et al.2003].
We can find more classical approaches like a rule-based system [Grazina et al.2011] or phrase-based and neural systems [Costa-jussà et al.2018] in translating between Brazilian Portuguese and European Portuguese. Also in Spanish-Portuguese a phrase-based system for broadcast news [Martínez et al.] or medical terms [Renato et al.2018].

With system combination, we could found a great variety of implementation like the CMU [Hildebrand and Vogel2009] or RWTH system combination [Leusch et al.2009], but in any of them, there is in consideration the usage in similar languages, where the phrase-based can offer more competitivity against neural.
Even though we found a similar approach in the work of Costa-Jussa, M.R. [Costa-jussà2017], where it's applied neural, rule and phrase-based systems in Catalan Spanish pair and use an MBR system combination, it takes some different ways as it doesn't use back-translation and so the system combination is not applied over them, and NMT uses an RNN with attention and doesn't add monolingual data.

In none of them, we found an application of a Neural MT or a system combination in the Spanish Portuguese or Czech Polish case.

5 System Combination with backtranslation

In order to achieve the best possible result in the translations, we propose to combine the results of both phrase-based and NMT systems at a sentence level so that we choose the better case for each of the sentences of the translated text. We aimed for a relatively simple combination strategy comparing with other previous work [Marie and Fujita2018].

The principle of this approach consists in the evaluation of the back-translations generated by both systems using the BLEU score [Papineni et al.2002], choosing the sentence that obtained better results. To obtain a score out of the translations texts, since. theoretically, we don't have a reference, we back-translated (translate again and obtain a text in the original language) each of the translations using both PB and NMT systems, instead of using only one, and weighted them equally. A graphical representation of this strategy can be found in Figure 9.

The final translated text is composed by each of the sentences from the system that obtained the highest score in the combined back-translation in each case.

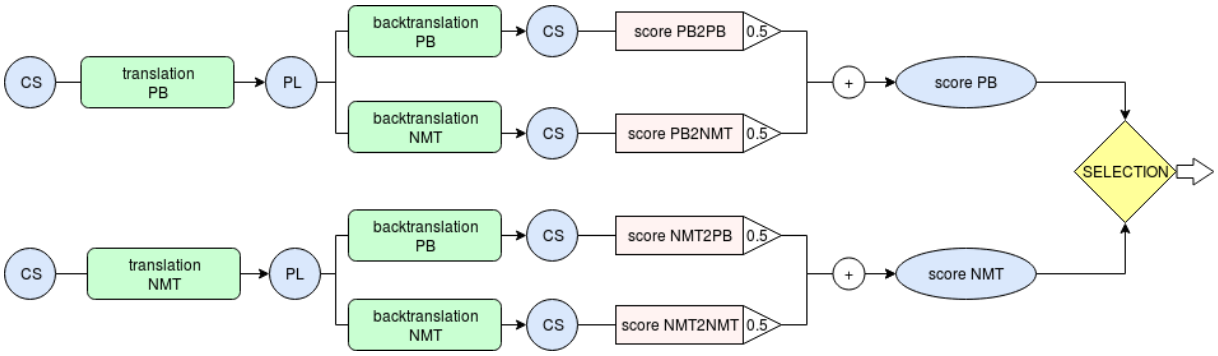

Figure 9: Back-translation selection approach

This approach is motivated by the recent success of different uses of back-translation in neural MT studies [Sennrich et al.2016b, Lample et al.2018].

Contrastive approaches We thought that to evaluate better the results obtained will be good to have some contrastive approaches, thus we decided to use MBR, which uses the translations instead of the back-translation, and length ratio, which doesn't consider the content of the sentences translated, as contrastive measures.

Minimum Bayes Risk [Kumar and Byrne2004] as said by Kumar, S. and Byrne, W. "consist in apply the techniques with the same name developed for automatic speech recognition [Goel and Byrne2000] and bitext word alignment for statistical MT [Kumar and Byrne2002] to the the problem of building automatic MT systems tuned for specific metrics". It aims for the solution that carries the least Bayesian risk since we are training and decoding with imperfect models.

The length ratio is a simple yet not content-based evaluation method since the premise is to calculate the ratio between the number of words in the input and output translated sentences. With this approach we are assuming that a good back-translation will be the one that contains the same numbers of words, regardless of which ones, as the source sentence.

6 Implementation

In this section, we report details about the data and preprocessing, the steps taken with Moses in order to configure the translation system and the parameter details of the systems and system combination.

6.1 Data and Preprocessing

Both systems, statistical and neural, use the corpora (monolingual or parallel) provided by the organizers, no external dataset is used. For both Czech-Polish and Spanish-Portuguese, we used all available parallel and monolingual data [table 6].

	idiom	sentences
parallel corpus	es-pt	2.481.441
	cz-pl	2.191.379
monolingual corpus	es	46.257.689
	pt	10.376.328
	cz	73.090.126
	pl	1.197.480
validation corpus	all	3.000
test corpus	es-pt	3.000
	cz-pl	3.412

Table 6: Number of sentences used

Regarding the validation set, we split it into two parts, the first, consisting of 2 thousand sentences, was used as additional training data, and the remaining part, consisting of 1 thousand sentences, was used as validation. Our test set corresponds to the official evaluation set.

The postprocessing of the test set once translated, was done in reverse order and included detruecasing and detokenization.

6.1.1 Moses

Moses allows to adjust its functioning in different ways and offers a great variety of functionalities. Although we implemented a bidirectional system, in this section I will explain the steps used in order to implement a unidirectional Spanish-Portuguese Phrase-based translation system with moses [12].

In this section, we will use the parameters $my_dir, as the personal directory of the user, and $moses_dir, as the ubication of Moses and all its tools. Also, we will use the following corpus:

[12] A more extensive documentation can be found in the Users Manual in http://www.statmt.org/moses/manual/manual.pdf

- the parallel corpus, as *corpusp*
- the monolingual and first 2k sentences of the validation will be merged in a sole corpus in order to have a bigger training corpus and improve the results, naming it *corpustraining*
- the remaining 1k sentences of the validation as a *dev1k* file
- a corpus combining all the other corpus named *corpusall*

Corpus Preprocessing

We have a folder "corpus" where all the corpus are stored and, as all we will do in this part will be in preprocessing level, all the steps taken in this subsection will be done being inside this directory.

```
cd $my_dir/corpus
```

- **Tokenize**:
  The first step'[13] is to normalize and tokenize, which means to separate the sentences in order to have every word and punctuation mark surrounded by spaces, all the sets of data that we will be using during the project.

  ```
  $moses_dir/scripts/tokenizer/normalize-punctuation.perl -l pt < \
  corpusp.pt | $moses_dir/scripts/tokenizer/tokenizer.perl -l pt > \
  corpusp.tok.pt
  ```

  The same process is needed for all the other corpus (corpustraining, dev1k and corpusall).

- **Truecase**:
  Lita, L.V. defines truecasing as "the process of restoring case information to badly-cased or non-cased text" [Lita et al.2003].

  In order to help Moses know which words should truecase we have to train a Truecaser. This Truecaser is global for all corpus and will have better results as more data we input, so we will use the corpusall corpus prepared before.

  ```
  $moses_dir/scripts/recaser/train-truecaser.perl -model \
  truecase-model.pt -corpus corpusall.tok.pt
  ```

  When we have the Truecaser trained, we truecase all the corpus except the corpusall, which from now on will no longer be needed.

  ```
  $moses_dir/script/recaser/truecase.perl -model truecase-model.pt < \
  corpusp.tok.pt > corpusp.tok.truecase.pt
  ```

[13] we take for granted that you have two correctly differentiated corpus, monolingual and parallel, for each of the two languages used in the translation

- **Cleaning**
  The last step in the preprocessing is to clean all corpus, which means to delete all the sentences wrong aligned or too large. We defined the maximum length of a sentence to be 50 words.

```
$moses_dir/scripts/training/clean-corpus-n.perl corpusp.tok.truecase \
es pt namecorpus.clean 1 50
```

  And with this, we finished the preprocessing of the data.

Language Model

To continue, we return to the original directory.

```
cd $my_dir
```

Now we have to generate the language model (explained in 3.1). Moses normally uses a language model based on the target language paralel corpus, but we added the training corpus in a new corpus named corpusmodel since using additional training data is often beneficial. Moses also gives different options to construct the Language Model such as RandLM, KenLM or OxLM. We used a 5-gram KenLM since is fast and use low memory.

We create a new directory to store all files related to the LM.

```
mkdir $my_dir/lm

$moses_dir/bin/lmplz -o 5 -T /tmp < corpusmodel.pt > lenguagemodel.arpa.pt
```

Once the model is done, we binarize it as this changes help to reduce loading time.

```
$moses_dir/bin/build_binary lenguagemodel.arpa.pt lenguagemodel.blm.pt
```

Even though is not strictly necessary, in our case we wanted to reduce the memory used to translate since we had some delays regarding the need of more memory, but it depends on the computer used and the size of the file to translate. In order to do that we applied two commands:

We tried to reduce the memory used by the LM. As a trade-off, we take more time to extract the LM but using less memory. This was accomplished doing:

```
bin/build_binary -a 64 trie languagemodel.arpa.pt languagemodel.blm.pt
```

We also used on-demand loading, in order to avoid loading the full LM into memory at the beginning. In order to do that we had to modify the line KenLM inside [features] from the moses.ini file created after the training, adding lazyken=true.

Training

Now we arrive at the training part. The command used consists of various functions such as word alignment in order to have an adequated corpus for Moses, phrase extraction and punctuation or the creation of the configuration file moses.ini. All of the halfway files and final configuration such as the moses.ini configuration file will be stored in a new directory named traines-pt.

```
mkdir $my_dir/traines-pt

$moses_dir/scripts/training/train-model.perl -root-dir $my_dir/traines-pt \
-corpus $my_dir/corpus/corpustraining.clean \
-f es -e pt  -external-bin-dir $moses_dir/tools -mgiza \
-alignment grow-diag-final-and \
-reordering msd-bidirectional-fe \
-lm 0:5:$my_dir/lm/lenguagemodel.blm.pt:8 -parallel > training.pt.out 2>&1
```

In our case, this part took 9 hours more or less to be finished, but again it depends on the computer capacity and the corpus size used.

Tuning

Finally, in order to calibrate and optimize the translation system, we apply the tunning, which in summary readjusts the word weights having in count the corpus used as validation, in our case the file dev1k. This part usually is the most time extensive one. In order to store all the files created during the tuning, a new "tuning" folder will be created inside the training folder.

```
mkdir $my_dir/traines-pt/tuning

$moses_dir/scripts/training/mert-moses.pl \
$my_dir/corpus/dev1k.clean.es $my_dir/corpus/dev1k.clean.pt \
$moses_dir/bin/moses $my_dir/traines-pt/model/moses.ini \
--working-dir $my_dir/traines-pt/tuning \
--nbest 100  -threads 16 \
--mertdir $moses_dir/bin/ --rootdir $moses_dir/scripts > \
$my_dir/traines-pt/mert.out 2>&1
```

Then, as it finished, we compacted the translation tables, which reduced by far the memory use.

```
$moses_dir/bin/processPhraseTableMin \
 -in $my_dir/traines-pt/model/phrase-table.gz \
 -out $my_dir/traines-pt/model/phrase-table \
 -nscores 4 -threads 4
sed 's,phrase-table.gz,phrase-table.minphr,g' -i \
$my_dir/traines-pt/tuning/moses.ini
sed 's,PhraseDictionaryMemory,PhraseDictionaryCompact,g' -i \
$my_dir/traines-pt/tuning/moses.ini
```

Translation

And we are done preparing the system to translate any text in Spanish to Portuguese with the instruction:

```
$moses_dir/bin/moses -f $my_dir/traines-pt/tuning/moses.ini \
-i inputfile.es > outputfile.pt
```

This implementation it's only for a unidireccional translation system, if a bidirection is wanted, it's necessary to repeat all the steps with the source and target languages being interchanged.

6.2 Neural System

The Neural system had two implementations, the first was implemented using the standard paralel corpus for training, but the second one was using a back-translated monolingual corpus as a suplementary data (as mentioned in section 3.2.1), referred as pseudocorpus.

Although I didn't participate directly in the Neural system implementation, the data used as pseudocorpus was translated using the phrase-based System. More specifically, the data back-translated to be used as pseudo-corpus was the monolingual corpus of the target language, in our case the Portuguese and Polish monolingual corpus. This files, as seen in table 6, contained a great quantity of sentences, and it took us a lot of time to translate them.

6.3 System combination

In order to implement the system combination with back-translation (explained in section 5) we used the BLEU score (section 3.3) in a sentence level with the *sentence-bleu* script available from Moses. We also gave the same weights W=1/2 (Fig 9) to both phrase and neural-based backtranslations. The wheights assigned to both phrase and neural-based were equals.

For the contrastive approach MBR we used an implementation available from Moses, and for the length ratio approach, we kept the translation with the ratio closer to 1.

In case of ties, we kept the sentence from the system that scored the best according to Table 7.

6.4 Parameters

6.4.1 Phrase-based

For the phrase-based systems we used Moses [Koehn et al.2007], which is a statistical machine translation engine, open source. In our case, in order to build it, we used in general the default parameters which include: grow-diagonal-final-and word alignment, lexical msd-bidirectional-fe reordering model trained, lexical weights, binarized and compacted phrase table with 4 score components and 4 threads used for conversion, 5-gram, binarized, loading-on-demand language model with Kneser-Ney smoothing and trie data structure without pruning; and MERT (Minimum Error Rate Training) optimisation with 100 nbestlist generated and 16 threads.

6.4.2 Neural-based

Our neural network model submission is based on the Transformer architecture (as described in section 1.3) implemented by Facebook in the `fairseq` toolkit [14]. The following hyperparameter configuration was used: 6 attention layers in the encoder and the decoder, with 4 attention heads per layer, embedding dimension equals 512, maximum number of tokens per batch set to 4000, Adam optimizer with $\beta_1 = 0.90$, $\beta_2 = 0.98$, varied learning rate with the inverse square root of the step number (warmup steps equal 4000), dropout regularization and label smoothing set to 0.1, weight decay and gradient clipping threshold set to 0.

[14] https://github.com/pytorch/fairseq

7 Results

This section will include findings and little analysis of the data collected.

The evaluations with BLEU score of the translations with both phrase-based and neural-based systems (*baseline systems*) can be found in the Table 7. With the NMT systems there are also two approaches using additional data: either using *monolingual* corpus on both source and target sides or using the back-translation with the phrase-based system to obtain the *pseudocorpus*, as explained in section 6.2. The first observation that can be made is the difference of values between the two pairs of languages although both been equally considered similar languages. An additional interesting observation can be how in both additional approaches with NMT, despite being very similar in concepts, the monolingual seems to harm the performance of the system, while the pseudo-corpus, only in the Czech-to-Polish case, improve the results obtained by the baseline system.

The two systems sent to the WMT task evaluation were the pseudo-corpus NMT system for CS-PL and the baseline PB system for ES-PT, which were ranked 1st and 2nd for their respectively submitted languages. The results obtained by the evaluators using the BLEU and TER evaluation metrics, as explained in 2, are in Table 8.

	CS-PL	**ES-PT**
PB	9.87	64.96
NMT	11.69	58.40
NMT+mono	10.91	52.37
NMT+pseudo corpus	12.76	–

Table 7: Phrase-based (PB) and Neural-based (NMT) results

System	**Language pair**	**BLEU**	**TER**
pseudo NMT	CS-PL	7.9	85.9
baseline PB	ES-PT	62.1	23.0

Table 8: Results in the WMT evaluation

In Table 9, we encounter the results of the back-translations. The back-translations were obtained from the translation with the best NMT system from Table 7. In both cases we can observe how the PB back-translation system that comes from the PB trasnlation surpass all the others with a considerable distance from them.

1st sys	2nd sys	**PL-CS**	**ES-PT**
PB	PB	44.34	84.62
	NMT	24.51	66.15
NMT	PB	32.47	63.37
	NMT	27.31	60.01

Table 9: Back-translation systems results

As presented in Table 10, our proposed system combinations, employing either MBR, back-translation or length ratio approach, did not achieve any significant improvements. The MBR strategy was applied for all systems from Table 7, which means a 4 systems combination in both pair languages.

	CS-PL	**ES-PT**
MBR	12.75	62.17
Backtranslation	10.73	64.97
Length Ratio	10.65	63.36

Table 10: System Combination Results

To further study the use of the back-translations combination, we implemented four modifications to the first back-translation approach, using part of the available back-translations:

- using only the PB (onlyPB) or NMT (onlyNMT) back-translations systems for both translations
- using the same back-translation system as the translation (corresponding) or the contrary (inverse)

The schemes of these new approaches are in Figure 10.

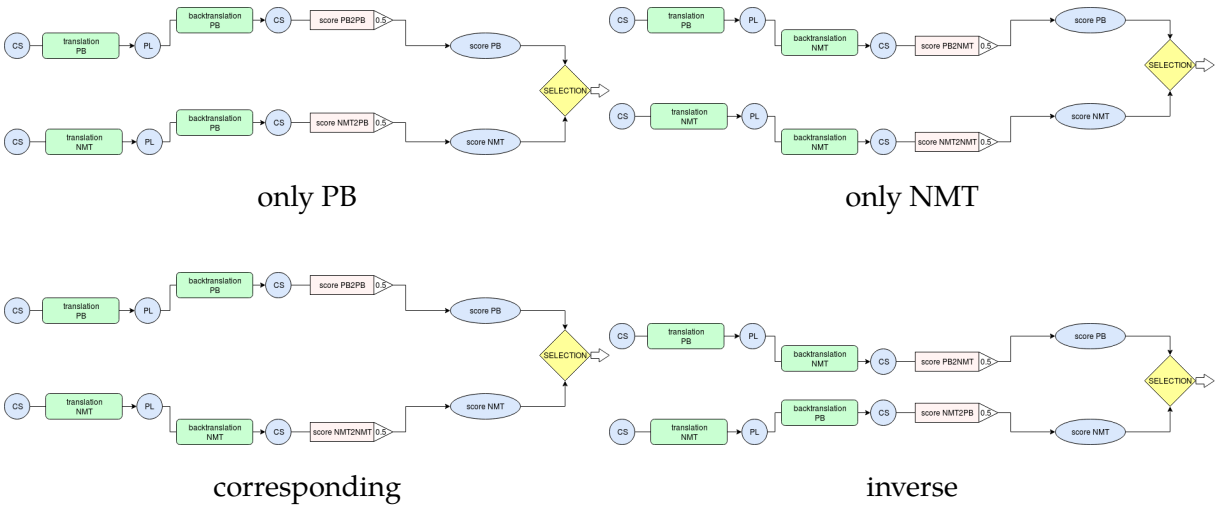

Figure 10: Different system combination approaches

In order to analyze better the reason behind the weak performance of the system combinations with back-translations, we evaluated the correlation, in both PB and NMT systems, between the quality of the translated sentence and the quality of the same sentence weighted back-translations.

In Table 11 we can find the results and how the low values obtained for any combination in both pairs, especially in the Czech-Polish case where this correlation varies between 0.15 and 0.3, could explain the poor performance of back-translation as a quality estimation metric.

		BLEU	correl PB	correl NMT
cs pl	normal combination	10.73	0.2473	0.2504
	onlyPB	10.30	0.1534	0.1930
	onlyNMT	11.20	0.2763	0.2391
	corresponding	10.35	0.1534	0.2391
	inverse	11.11	0.2763	0.1930
es pt	normal combination	64.97	0.4040	0.6542
	onlyPB	64.60	0.3424	0.6221
	onlyNMT	63.90	0.3403	0.5870
	corresponding	65.44	0.3424	0.5870
	inverse	61.73	0.3403	0.6221

Table 11: Correlations and BLEUs for the various combinations.

Also can be detected that both systems act in an opposite way. When we use the back-translation PB system in cs-pl, worse correlation values are obtained than if we used the back-translation NMT system, on the contrary with the es-pt systems, worse correlation values are obtained with NMT than with PB systems, in both cases coincide that the worst correlation is obtained by same system that get the worst score.
To check if the previous expressions are true, we calculated as well (in Table 12) for both pair of languages and both PB and NMT translation systems the correlation in quality between the translation and the back-translations from both back-translation systems. Certainly, we can observe how the results with the PB systems for CS-PL and NMT systems for ES-PT are less correlated.

correlation	PB2PB	PB2NMT	NMT2PB	NMT2NMT
cspl	0.1534	0.27623	0.1930	0.2391
espt	0.3424	0.3403	0.6221	0.5870

Table 12: Correlation between translation and back-translations

8 Conclusions and Further Research

In this section we discuss the main findings during our research.

We were surprised that translation performance was much lower from Czech-Polish than for Spanish-Portuguese. Since both tasks involve languages from the same family, we expected similar results. As reported in our paper [Biesialska et al.2019], we performed some hypothesis to explain these low performance. We know that Czech and Polish have some things in common. They are two languages considered similar that share the Western Slavic subgroup within the same language family, the Slavic, and have some common characteristics such as 7 noun cases, 2 number cases, 3 noun gender cases as well as 3 tenses among others. But still, similar languages tend to have quite in common, and, in part, thanks to this resemblance, obtain higher BLEU scores than other more distant pairs of languages.

Gamallo proposed a metric based on the perplexity as a measure of the distance between languages [Gamallo et al.2017] which could help us to understand this difference in score. In the distances between languages obtained with this metric (Table 13) we can observe how the distance between Slavic languages is generally higher than between Romance languages, being the Polish case the one that gets worse values within the Slavic group, obtaining a distance of 27 in the Czech-Polish case, while the Spanish-Portuguese is one of the bests with a distance of only 7.

Slavic		Latin		Mix	
pair	distance	pair	distance	pair	distance
cs-pl	27	es-pt	7	es-cs	37
cs-sl	8	es-fr	15	es-pl	44
cs-ru	21	es-ro	20	pt-cs	31
pl-sl	24	pt-fr	15	pt-pl	38
pl-ru	34	pt-ro	22		

Table 13: Language Distances within some Slavic, Romance and across languages families.

Despite the scant difference of number of letters in the alphabet (Table 1), unlike Spanish and Portuguese, we hypothesised that, even though both Czech and Polish languages come from the same origin, we can find in the diacritics an important characteristic when it comes to influencing the results considering that the difference in diacritics used in both languages could be considered significant:

$ą, ć, ę, ł, ń, ó, ś, ź, ż$ in Polish

$á, č, ď, é, ě, ch, í, ň, ó, ř, š, ť, ú, ů, ý, ž$ in Czech

Having in mind the diacritics, Czech language consists of 42 unique letters, while Polish is constituted by 32. Moreover, some of the letters that don't appear in the alphabets of both languages are used only in case of foreign words, that's the case for q, x w for Czech, and q, x v for Polish.

In our concrete case the PB system offers better results compared with the NMT system in case of similar languages with very low distance, whereas with more distanced languages, NMT will have a better performance. In fact, we can't draw final conclusions from this correlation, but it could be analyzed with more attention as future research.

Another point to comment is how, just like had been discussed in other works [Somers2005], a back-translation system is not a good metric to evaluate a translation system.

Back-translation doesn't work for various reasons, just as O'Connell [O'Connell2001] and other authors commentated on: it could be due to a bad functioning of the back-translation system, which in this case you can't differentiate if it stems from a bad translation or a bad back-translation. In fact, and as the second point, it could be the case where, despite having a bad translation, you could obtain a good result from the back-translation. In other words, a high score from back-translation doesn't mean a good translation, this is why we obtained that low correlation results in Tables 11 and 12.

Finally, even experienced human translators don't expect to achieve an identical translation word by word, but that's a thing that is penalized when using automatic evaluation metrics, thus a good evaluation when using back-translation could be using human evaluation, but that would mean an expense on time and resources normally to big to be considered. This problem impedes us to do a good selection in order to improve the results using the combination of systems. Other automatic metrics performance could be examined, using more features for instance [Marie and Fujita2018], as future research.

9 Appendix

9.1 Costs

In this section, we take into account the cost of the project. We considered both Magdalena and myself as a Teamworkers and supervisor Marta Costa-Jussa as the Supervisor.

For the salaries, we assumed that both Teamworkers work the same hours per week, 20 hours, and the project term is 20 weeks. As the Leader main task was the supervision and she was involved in other projects, she didn't work in it the same number of hours as the Teamworkers, 8 hours per week for the full duration. We have to include the social security costs payment, which is a 33.4% rate.

Description	Quantity	€/hour	€/week	Social Security	Cost
Supervisor	1	40	320	2.137,6	8.537,6
Teamworkers	2	25	1.000	6.680	26.680
Total cost					**35.217,6**

Table 14: Total cost of the salaries

As office expenses, we needed an office. The cost for an already furnished office near our campus is 500€/month, we rented it for 5 months. So, the total cost 2500€.
Additionally, our team needed powerful computers to develop the project, one for each one, to work simultaneously. The cost of these computers is approximately $\frac{3*700*0.9}{5} = 378$ for a year, and as we used them for 5/12 of the year $\frac{375*12}{5} = 158$.

Description	Quantity	€/unit	Useful life	Cost
Computer	3	700	5	158
Description		**€/month**	**months**	**Costs**
Rent office		500	5	2.500
Total cost				**2.658**

Table 15: Office expenses cost

As explained in section 1.3, our product was implemented using Fairseq and Moses toolkits, which are open-source. The coding of the project was written using Bash scripts. No license was necessary for any of the programs used.

As electricity consumption, we have to take in count the consumption of the office and the computers. We hired Endesa One Luz rate (0.12€/kWh).
As the electricity consumption of the office during the time we used it, is about 30 kWh approximately. The previous calculations include the electricity generated in the laboratory by the lights and other electronic devices but not the computers.
As computer electric consumption, computers use an average of 72 kWh of energy consumption per computer. This sums up to:

$$(72kWh * 3computers + 30kWh) * 0.12€/kWh = 29,52€/month \quad (6)$$

Description	€/month	months	cost
electricity	29,52	5	147,60
Total cost			**147,60**

Table 16: Final cost consumption of electricity

Finally, the sum of all the various costs concludes in a total of 38.022,70€, as shown in Table 17.

Description	Cost
Salaries	35.217,60
Office	2.658
Supplies	147,60
Total cost	**38.023,20**

Table 17: Final cost of the project

9.1.1 Environmental cost

Our product is not material; Consequently, the environmental impact we produce is reduced as there is no need to deal with any potentially harmful substances or exploitation of resources. However, we have to take into account the amount of impact caused by the electricity usage.

We consumed a quantity of:

$$\left(2computers * 72kWh + 30kWh\right) * 400h + 1computer * 72kWh * 160h = 81.120kW \quad (7)$$

Taking into account a generation of CO^2 per electricity consumption of $0,649kgCO^2/kWh$ [15]. And with consumption of 81.120 kWh, we generate:

$$\text{Total kg } CO^2/\text{project} = 0,649kg*CO^2/kWh * 81.120kWh = 52.646,88kg\ CO^2 \quad (8)$$

[15] proposed by IDAE and described in the CALENER GT document, section 3.6

9.2 WMT submission

The following pages include the paper accepted at the Fourth Conference on Machine Translation (WMT19).

The TALP-UPC System for the WMT Similar Language Task: Statistical vs Neural Machine Translation

Magdalena Biesialska **Lluis Guardia** **Marta R. Costa-jussà**
TALP Research Center, Universitat Politècnica de Catalunya, 08034 Barcelona
magdalena.biesialska@upc.edu lluis.guardia@alu-etsetb.upc.edu
marta.ruiz@upc.edu

Abstract

Although the problem of similar language translation has been an area of research interest for many years, yet it is still far from being solved. In this paper, we study the performance of two popular approaches: statistical and neural. We conclude that both methods yield similar results; however, the performance varies depending on the language pair. While the statistical approach outperforms the neural one by a difference of 6 BLEU points for the Spanish-Portuguese language pair, the proposed neural model surpasses the statistical one by a difference of 2 BLEU points for Czech-Polish. In the former case, the language similarity (based on perplexity) is much higher than in the latter case. Additionally, we report negative results for the system combination with back-translation.

Our TALP-UPC system submission won 1st place for Czech→Polish and 2nd place for Spanish→Portuguese in the official evaluation of the 1st WMT Similar Language Translation task.

1 Introduction

Much research work has been done on language translation in the past decades. Given recent success of various machine translation (MT) systems, it is not surprising that some could consider similar language translation an already solved task. However, there are still remaining challenges that need to be addressed, such as limited resources or out-of-domain. Apart from these well-known, standard problems, we have discovered other under-researched phenomena within the task of similar language translation. Specifically, there exist languages from the same linguistic family that have a high degree of difference in alphabets, as it is the case for Czech-Polish, which may pose a challenge for MT systems.

Neural MT has achieved the best results in many tasks, outperforming former statistical MT (SMT) methods (Sennrich et al., 2016a). However, there are tasks where previous statistical MT approaches are still competitive, such as unsupervised machine translation (Artetxe et al., 2018; Lample et al., 2018). Motivated by the close proximity between the languages at hand and limited resources, in this article we aimed to determine whether the neural or the statistical approach is a better one to solve the given problem.

We report our results in the 1st Similar Language Translation WMT task (Barrault et al., 2019). In the official evaluation, our Czech→Polish and Spanish→Portuguese translation systems were ranked 1st and 2nd respectively. The main contributions of our work are the neural and statistical MT systems trained for similar languages, as well as the strategies for adding monolingual corpora in neural MT. Additionally, we report negative results on the system combination by using back-translation and Minimum Bayes Risk (Kumar and Byrne, 2004) techniques.

2 Background

In this section, we provide a brief overview of statistical (phrase-based) and neural-based MT approaches as well as strategies for exploiting monolingual data.

2.1 Phrase-based Approach

Phrase-based (PB) statistical MT (Koehn et al., 2003) translates by concatenating at a phrase level the most probable target given the source text. In this context, a phrase is a sequence of words, regardless if it is a phrase or not from the linguistic point of view. Phrases are extracted from word alignments between both languages in a large parallel corpus, based on the probabilistic study, which identifies each phrase with several features,

such as conditional probabilities. The collection of scored phrases constitutes the translation model.

In addition to this model, there are also other models to help achieve a better translation, such as the reordering model, which helps in a better ordering of the phrases; or the language model, trained from a monolingual corpus in the target language helping to obtain a better fluency in the translation. The weights of each of these models are optimized by tuning over a validation set. Based on these optimized combinations, the decoder uses beam search to find the most probable output given an input. Figure 1 shows a diagram of the phrase-based MT approach.

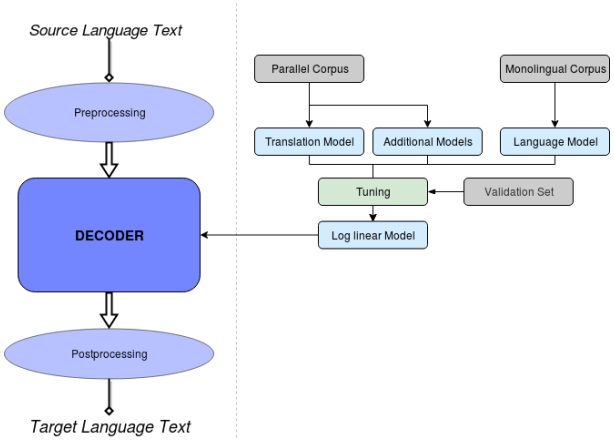

Figure 1: Basic schema of a phrase-based MT system

2.2 Neural Approach

Neural networks (NNs) have been successful in many Natural Language Processing (NLP) tasks in recent years. NMT systems, which use end-to-end NN models to encode a source sequence in one language and decode a target sequence in the second language, early on demonstrated performance on a par with or even outperformed traditional phrase-based SMT systems (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015; Sennrich et al., 2016a; Zhou et al., 2016; Wu et al., 2016).

Previous state-of-the-art NMT models used predominantly bi-directional recurrent neural networks (RNN) equipped with Long-Short Term Memory (LSTM; Hochreiter and Schmidhuber 1997) units or Gated Recurrent Units (GRU; Cho et al. 2014) both in the encoder and the decoder combined with the attention mechanism (Bahdanau et al., 2015; Luong et al., 2015). There were also approaches, although less common, to leverage convolutional neural networks (CNN) for sequence modeling (Kalchbrenner et al., 2016; Gehring et al., 2017).

In this work, we focus on the most current state-of-the-art NMT architecture, the Transformer (Vaswani et al., 2017), which shows significant performance improvements over traditional sequence-to-sequence models. Interestingly, while the Transformer employs many concepts that were used earlier in encoder-decoder RNN and CNN based models, such as: residual connections (He et al., 2016b), position embeddings (Gehring et al., 2017), attention; the Transformer architecture relies solely on the self-attention mechanism without resorting to either recurrence or convolution.

The variant of the self-attention mechanism implemented by the Transformer, multi-head attention, allows to model dependencies between all tokens in a sequence irrespective of their actual position. More specifically, the representation of a given word is produced by means of computing a weighted average of attention scores of all words in a sentence.

Adding Monolingual Data Although our proposed statistical MT model incorporates monolingual corpora, the supervised neural MT approach is not capable to make use of such data. However, recent studies have reported notable improvements in the translation quality when monolingual corpora were added to the training corpora, either through back-translation (Sennrich et al., 2016b) or copied corpus (Currey et al., 2017). Encouraged by those results and given the similarity of languages at hand, we propose to exploit monolingual data by leveraging back-translation as well as by simply copying target-side monolingual corpus and use it together with the original parallel data.

3 System Combination with Back-translation

In this paper, we propose to combine the results of both phrase-based and NMT systems at the sentence level. However, differently from the previous work of Marie and Fujita (2018), we aimed for a conceptually simple combination strategy.

In principle, for every sentence generated by the two alternative systems we used the BLEU score (Papineni et al., 2002) to select a sentence with the highest translation quality. Each of the translations was back-translated (i.e. translated from the target language to the source language). In-

stead of using only one system to perform back-translation, we used both PB and neural MT systems and weighted them equally. See Figure 2 for a graphical representation of this strategy.

This approach was motivated by the recent success of different uses of back-translation in neural MT studies (Sennrich et al., 2016b; Lample et al., 2018). The final test set was composed of sentences produced by the system that obtained the highest score based on the quality of the combined back-translation.

4 Experimental Framework

In this section we describe the data sets, data preprocessing as well as training and evaluation details for the PB and neural MT systems and the system combination.

4.1 Data and Preprocessing

Both submitted systems are constrained, hence they don't use any additional parallel or monolingual corpora except for the datasets provided by the organizers. For both Czech-Polish and Spanish-Portuguese, we used all available parallel training data, which in the case of Czech-Polish consisted of about 2.2 million sentences and about 4.5 million sentences in the case of Spanish-Portuguese. Also, we used all the target-side monolingual data, which was 1.2 million sentences for Polish and 10.9 million sentences for Portuguese.

Preprocessing Our NMT model was trained on a combination of the original Czech-Polish parallel corpus together with pseudo-parallel corpus obtained from translating Polish monolingual data to Czech with Moses. Additionally, the development corpus was split into two sets: first containing 2k sentences and second containing 1k sentences, where the former was added to the training data and the latter was used for validation purposes.

Our Phrase-Based model was trained on a combination of the original Spanish-Portuguese parallel corpus together with 2k sentences from the dev corpus. Specifically, the development corpus was split into two sets: first containing 2k sentences and second containing 1k sentences, where the former was added to the training data and the latter was used for validation purposes.

Then we followed the standard preprocessing scheme, where training, dev and test data are normalized, tokenized and truecased using *Moses*[1] scripts. Additionally, training data was also cleaned with `clean-corpus-n.perl` script from *Moses*. Finally, to allow open-vocabulary, we learned and applied byte-pair encoding (BPE)[2] for the concatenation of the source and target languages with 16k operations. The postprocessing was done in reverse order and included detruecasing and detokenization.

4.2 Parameter Details

Phrase-based For the Phrase-based systems we used Moses (Koehn et al., 2007), which is a statistical machine translation system. In order to build our model, we used generally the default parameters which include: grow-diagonal-final-and word alignment, lexical msd-bidirectional-fe reordering model trained, lexical weights, binarized and compacted phrase table with 4 score components and 4 threads used for conversion, 5-gram, binarized, loading-on-demand language model with Kneser-Ney smoothing and trie data structure without pruning; and MERT (Minimum Error Rate Training) optimisation with 100 n-best list generated and 16 threads.

Neural-based Our neural network model is based on the Transformer architecture (as described in section 2.2) implemented by Facebook in the *fairseq* toolkit[3]. The following hyperparameter configuration was used: 6 attention layers in the encoder and the decoder, with 4 attention heads per layer, embedding dimension of 512, maximum number of tokens per batch set to 4000, Adam optimizer with $\beta_1 = 0.90$, $\beta_2 = 0.98$, varied learning rate with the inverse square root of the step number (warmup steps equal 4000), dropout regularization and label smoothing set to 0.1, weight decay and gradient clipping threshold set to 0.

System Combination The key parameter in the system combination with back-translation, explained in section 3, is the score. Hence, we used the BLEU score (Papineni et al., 2002) at the sentence level, implemented as *sentence-bleu* in *Moses*. Furthermore, we assigned equal weights to both phrase and neural-based translations and back-translations.

[1] https://github.com/moses-smt/mosesdecoder
[2] https://github.com/rsennrich/subword-nmt
[3] https://github.com/pytorch/fairseq

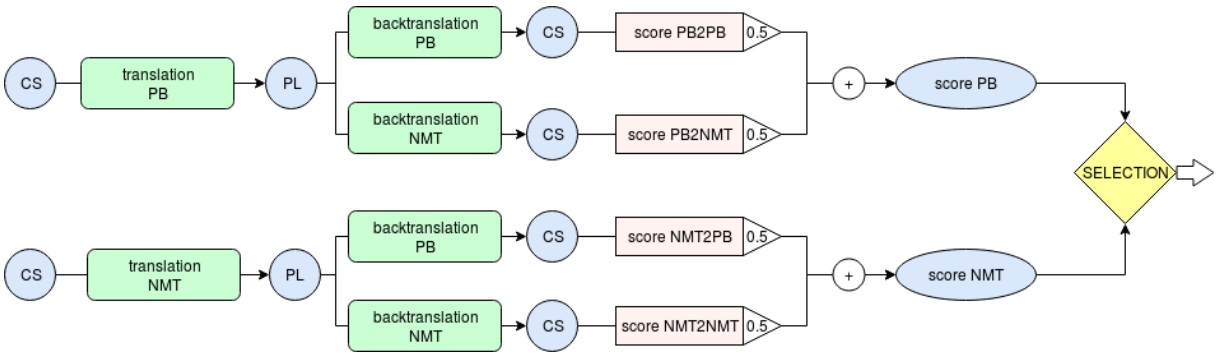

Figure 2: Scheme of the system combination approach

As contrastive approaches for system combination, we used two additional strategies: Minimum Bayes Risk (Kumar and Byrne, 2004) and the length ratio. In the former case, we used the implementation available in *Moses*. In the latter approach, the ratio was computed as the number of words in the translation divided by the number of words in the source input. Sentence translations that gave a length ratio closer to 1 were selected. In the case of ties, we kept the sentence from the system that scored the best according to Table 3.

5 Results

The results provided in Table 1 show BLEU scores for the direct phrase-based and neural-based MT systems. Also, we report on experiments with incorporating monolingual data in two ways: either using a monolingual corpus both on the source and target sides (*monolingual*) or using the back-translation system to produce a translation of a monolingual corpus (*pseudo corpus*). Interestingly, we observe that the *monolingual* approach harms the performance of the system even in the case of similar languages. With regard to the Spanish-Portuguese language pair, due to the large size of the monolingual corpora as well as the time constraint, we were unable to finish training of our NMT model with the pseudo corpus.

Table 1: Phrase-based (PB) and Neural-based (NMT) results.

	CS-PL	**ES-PT**
PB	9.87	64.96
NMT	11.69	58.40
NMT + monolingual	10.91	52.37
NMT + pseudo corpus	12.76	–

As presented in Table 3, our proposed system combinations, employing either MBR or the back-translation approach, did not achieve any significant improvements. The MBR strategy was applied to all systems from Table 1, which means that for the Czech-Polish pair we used 4 systems and for Spanish-Portuguese we used 3 systems. Back-translation results were evaluated with respect to the systems in Table 2 and the system combination with back-translation was created using the best two systems from Table 1.

Table 2: Back-translation system results.

1st system	2nd system	**PL-CS**	**PT-ES**
PB	PB	44.34	84.62
	NMT	24.51	66.15
NMT	PB	32.47	63.37
	NMT	27.31	60.01

Table 3: System Combination results.

	CS-PL	**ES-PT**
MBR	12.75	62.17
Back-translation	10.71	64.97

In order to analyze the reason behind the weak performance of the system combination with back-translation, we evaluated the correlation between the quality of each translated sentence (generated using PB and NMT systems) and the quality of back-translations (both for PB and NMT systems) on the validation set. For any combination, Czech-Polish or Spanish-Portuguese, correlation varies between 0.2 and 0.4, which explains the poor performance of back-translation as a quality estimation metric.

6 Discussion

Although Czech and Polish belong to the same family of languages (Slavic) and share the same subgroup (Western Slavic), the BLEU score obtained by our winning system is relatively low comparing to other pairs of similar languages (e.g. Spanish and Portuguese). It may seem surprising considering some common characteristics shared

by both languages, such as 7 noun cases, 2 number cases, 3 noun gender cases as well as 3 tenses among others.

Low performance on this task could be explained by the language distance. Considering the metric proposed by Gamallo et al. (2017), which is based on perplexity as a distance measure between languages, the distance between Czech and Polish is 27 while for Spanish-Portuguese is 7. The very same metric used to evaluate the distance of Czech and Polish from other Slavic languages (i.e. Slovak and Russian) shows that Polish is the most distant language within this group (see Table 4). In general, distances between Latin languages are smaller than between Slavic ones.

Table 4: Distances between Slavic and Latin languages. Examples across families.

Slavic		Latin		Mix	
pair	dist.	pair	dist.	pair	dist.
CS-PL	27	ES-PT	7	ES-CS	37
CS-SL	8	ES-FR	15	ES-PL	44
CS-RU	21	ES-RO	20	PT-CS	31
PL-SL	24	PT-FR	15	PT-PL	38
PL-RU	34	PT-RO	22		

While Czech and Polish languages are highly inflected, which poses a challenge, we hypothesize that one of the reasons for the low BLEU score lies also in the difference of the alphabets. Even though both alphabets are based on the Latin script, they include letters with diacritics – *ą, ć, ę, ł, ń, ó, ś, ź, ż* in Polish, and *á, č, d', é, ě, ch, í, ň, ó, ř, š, t', ú, ů, ý, ž* in Czech. The total number of unique letters in Polish is 32, while in the Czech language there are 42 letters. Moreover, some letters are used only in the case of foreign words, such as *q*, *x* (in Czech and Polish), *w* (in Czech), and *v* (in Polish).

7 Future Work

In the future we plan to extend our research in the following directions. First, we would like to explore how removing diacritics on the source-side would impact the performance of our system for the Czech-Polish language pair. Furthermore, we would like to study the performance of our system combination while applying various quality estimation approaches. We would be interested in experimenting with the reward score introduced by He et al. (2016a), which is a linear combination of language model score and the reconstruction probability of the back-translated sentence, as well as with other quality measures implemented in the *OpenKiwi* (Kepler et al., 2019) toolkit[4].

Acknowledgments

The authors want to thank Pablo Gamallo, José Ramom Pichel Campos and Iñaki Alegria for sharing their valuable insights on their language distance studies.

This work is supported in part by the Spanish Ministerio de Economía y Competitividad, the European Regional Development Fund and the Agencia Estatal de Investigación, through the postdoctoral senior grant Ramón y Cajal, the contract TEC2015-69266-P (MINECO/FEDER,EU) and the contract PCIN-2017-079 (AEI/MINECO).

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation, Volume 2: Shared Task Papers*, Florence, Italy. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar*, pages 1724–1734.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, Copenhagen, Denmark. Association for Computational Linguistics.

[4]https://github.com/Unbabel/OpenKiwi

Pablo Gamallo, José Ramom Pichel, and Iñaki Alegria. 2017. From language identification to language distance. *Physica A: Statistical Mechanics and its Applications*, 484:152 – 162.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1243–1252.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016a. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016b. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709. Association for Computational Linguistics.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aäron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural machine translation in linear time. *CoRR*, abs/1610.10099.

Fábio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André FT Martins. 2019. Openkiwi: An open source framework for quality estimation. *arXiv preprint arXiv:1902.08646*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. In *HLT-NAACL 2004: Main Proceedings*, pages 169–176, Boston, Massachusetts, USA. Association for Computational Linguistics.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Benjamin Marie and Atsushi Fujita. 2018. A smorgasbord of features to combine phrase-based and neural machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 111–124, Boston, MA. Association for Machine Translation in the Americas.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation*, pages 371–376, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto

Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Jie Zhou, Ying Cao, Xuguang Wang, Peng Li, and Wei Xu. 2016. Deep recurrent models with fast-forward connections for neural machine translation. *TACL*, 4:371–383.

10 Bibliography

[Alonso2005] J.A. Alonso. 2005. Machine Translation for Catalan Spanish : The real case for productive MT.

[Bahdanau et al.2014] D. Bahdanau, C. Kyunghyun, and Y. Bengio. 2014. Neural machine translation by jointly learning to align and translate.

[Barzilai and Borwein1988] J. Barzilai and J. M. Borwein. 1988. Two-Point Step Size Gradient Methods. *IMA Journal of Numerical Analysis*, 8(1):141–148.

[Becher1962] Johann Joachim Becher, 1962. *Zur mechanischen Sprachübersetzung. Ein Programmierversuch aus dem Jahre 1661*.

[Biesialska et al.2019] Magdalena Biesialska, Lluis Guardia, and Marta R. Costa-jussà. 2019. The talp-upc system for the wmt similar language task: Statistical vs neural machine translation. In *Proceedings of the 4th Conference on Machine Translation*, Florence, August. Association for Computational Linguistics.

[Bojar et al.2016] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. J. Yepes, P. Koehn, V. Logacheva, C. Monz, M. Negri, A. Névéol, M. Neves, M. Popel, M. Post, R. Rubino, C. Scarton, L. Specia, M. Turchi, K. Verspoor, and M. Zampieri. 2016. Findings of the 2016 Conference on Machine Translation. In *ACL 2016 First Conference on Machine Translation (WMT16)*, pages 131–198. The Association for Computational Linguistics.

[Brown et al.1990] P.F. Brown, J. Cocke, S.A. Della Pietra, Della Pietra V.J., F. Jelinek, J.D. Lafferty, R.L. Mercer, and P.S. Roossin. 1990. A Statistical Approach To Machine Translation, Computational Linguistics. volume 16, pages 79–85.

[Brown et al.1993] P.F. Brown, Della Pietra V.J., S.A. Della Pietra, and R.L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. In *Computational Linguistics*, volume 19, pages 263–311.

[Bémová and Kubon1990] A. Bémová and V. Kubon. 1990. Czech-to-Russian Transducing Dictionary. pages 314–316, 01.

[Canals et al.2019] R Canals, A. Esteve, A. Garrido, M.I. Guardiola, A. Iturraspe, S. Montserrat, S. Ortiz, H. P. Pina, P.M. Perez, and M.L. Forcada. 2019. The Spanish Catalan machine translation system interNOSTRUM. pages 73–76, 06.

[Chiang2007] David Chiang. 2007. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2):201–228.

[Cho et al.2014] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[Costa-jussà et al.2018] M.R. Costa-jussà, M. Zampieri, and S. Pal. 2018. A Neural Approach to Language Variety Translation. *CoRR*, abs/1807.00651.

[Costa-jussà2017] M.R. Costa-jussà. 2017. Why Catalan-Spanish Neural Machine Translation? Analysis, comparison and combination with standard Rule and Phrase-based technologies. In *VarDial*.

[Coughlin2003] D. Coughlin. 2003. Correlating automated and human assessments of machine translation quality. In *In Proceedings of MT Summit IX*, pages 63–70.

[Dorr et al.1998] B. J. Dorr, P.W. Jordan, and J.W. Benoit. 1998. A survey of current paradigms in machine translation. Technical Report ALAMP-TR-027, UMIACS-TR-98-72, CS-TR-3961, College Park: University of Maryland.

[Gamallo et al.2017] P. Gamallo, J.R. Pichel, and I. Alegria. 2017. From language identification to language distance. *Physica A: Statistical Mechanics and its Applications*, 484:152 – 162.

[Garrido-Alenda et al.2003] A. Garrido-Alenda, P. C. Gilabert-zarco, J. A. Perez-ortiz, A. Pertusa, G. Ramirez-Sanchez, F. Sánchez-Martínez, M. A. Scalco, and M. L. Forcada. 2003. Shallow Parsing for Portuguese–Spanish Machine Translation. 11.

[Gehring et al.2017] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y.N. Dauphin. 2017. Convolutional Sequence to Sequence Learning. *CoRR*, abs/1705.03122.

[Goel and Byrne2000] V. Goel and W.J. Byrne. 2000. Minimum Bayes-risk automatic speech recognition. *Computer Speech Language*, 14(2):115 – 135.

[Grazina et al.2011] N. Grazina, W. Ling, and I. Trancoso. 2011. BP2EP - adaptation of Brazilian Portuguese texts to European Portuguese. In *In Proceedings of the 15th Conference of the European Association for Machine Translation, EAMT '11*, pages 129–136.

[Hajič et al.2000] J. Hajič, J. Hric, and V. Kuboň. 2000. Machine translation of very close languages. pages 7–12, 04.

[He et al.2015] K. He, X. Zhang, S. Ren, and J. Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR*, abs/1512.03385.

[Hildebrand and Vogel2009] AA.S. Hildebrand and S. Vogel. 2009. CMU System Combination for WMT'09. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, StatMT '09, pages 47–50, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Hochreiter1997] J. Hochreiter, S. Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, November.

[Kalchbrenner and Blunsom2013] N. Kalchbrenner and P. Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709. Association for Computational Linguistics.

[Kalchbrenner et al.2016] N. Kalchbrenner, L. Espeholt, K. Simonyan, A. van den Oord, A. Graves, and K. Kavukcuoglu. 2016. Neural Machine Translation in Linear Time. *CoRR*, abs/1610.10099.

[Kirschner1987] Z Kirschner. 1987. Apac3-2: An english-to-czech machine translation system. 01.

[Klein et al.2018] G. Klein, Y. Kim, Y. Deng, V. Nguyen, J. Senellart, and A.M. Rush. 2018. OpenNMT: Neural Machine Translation Toolkit. In *Proceedings of AMTA 2018*, volume 1, pages 177–184, March.

[Kneser and Ney1995] R. Kneser and H. Ney. 1995. Improved backing-off for M-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, pages 181–184. IEEE. In: Detroit, MI, USA.

[Koehn et al.2003] Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

[Koehn et al.2007] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

[Kumar and Byrne2002] S. Kumar and W. Byrne. 2002. Minimum Bayes-Risk Word Alignments of Bilingual Texts. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 140–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Kumar and Byrne2004] Shankar Kumar and William Byrne. 2004. Minimum Bayes-Risk Decoding for Statistical Machine Translation. In *HLT-NAACL 2004: Main Proceedings*, pages 169–176, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

[Lample et al.2018] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. A. Ranzato. 2018. Phrase-Based & Neural Unsupervised Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium, October-November. Association for Computational Linguistics.

[Lee1997] A. Gladwin Lee. 1997. Alan turing, enigma, and the breaking of german machine ciphers in world war ii.

[Leira] V. Leira. Alphabets, Letters and Diacritics in European Languages (as they appear in Geography).

[Leusch et al.2009] G. Leusch, E. Matusov, and H. Ney. 2009. The RWTH System Combination System for WMT 2009. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, StatMT '09, pages 51–55, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Lita et al.2003] L.V. Lita, A. Ittycheriah, S. Roukos, and N. Kambhatla. 2003. tRuEcasIng. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 41 of *ACL '03*, pages 152–159, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Luong et al.2015] T. Luong, H.Q. Pham, and C.D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

[Marie and Fujita2018] B. Marie and A. Fujita. 2018. A Smorgasbord of Features to Combine Phrase-Based and Neural Machine Translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 111–124, Boston, MA, March. Association for Machine Translation in the Americas.

[Martínez et al.] R. S. Martínez, J. P. da Silva, and D. A. Caseiro. "Statistical Machine Translation of Broadcast News from Spanish to Portuguese. In *Computational Processing of the Portuguese Language", year="2008*, pages 112–121, Berlin, Heidelberg. Springer Berlin Heidelberg.

[McCulloch and Pitts1943] W.S. McCulloch and W. Pitts. 1943. A logical calculus of ideas immanent in nervous activity. In *Bulletin of Mathematical Biophysics*, volume 5.

[Navarro Cerdán et al.2004] J. Navarro Cerdán, J. González, D. Picó, F. Casacuberta, J. Val, F. Fabregat, F. Pla, and J. Tomás. 2004. Sishitra : A hybrid machine translation system from spanish to catalan. volume 3230, pages 349–359, 01.

[Och et al.1995] F.J. Och, C. Tillman, and H. Ney. 1995. Improved alignment models for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, June. In: University of Maryland,College Park, MD.

[O'Connell2001] T.A. O'Connell. 2001. Preparing your website for machine translation: how to avoid losing (or gaining)something in the translation. IBM website.

[Papineni et al.2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

[Renato et al.2018] A. Renato, J. Castaño, M. A. Williams, H. Berinsky, M. Gambarte, H. Park, D. Pérez, C. Otero, and D. Luna. 2018. A Machine Translation Approach for Medical Terms. In *InProceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2018)*, volume 5: HEALTHINF, pages 369–378, 01.

[Rosenblatt1961] F. Rosenblatt. 1961. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*.

[Scannell2006] K. Scannell. 2006. Machine translation for closely related language pairs. *Proceedings of the LREC 2006 Workshop on Strategies for Developing Machine Translation for Minority Languages*, 01.

[Sennrich et al.2016a] R. Sennrich, B. Haddow, and A. Birch. 2016a. Edinburgh Neural Machine Translation Systems for WMT 16. In *Proceedings of the First Conference on Machine Translation*, pages 371–376, Berlin, Germany, August. Association for Computational Linguistics.

[Sennrich et al.2016b] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August. Association for Computational Linguistics.

[Snover et al.2006] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231.

[Somers2005] H. Somers. 2005. Round-trip translation: what is it good for? In *Australasian Language Technology Workshop 2005 (ALTW 2005): Proceedings of the Workshop*, pages 10–11. University of Sydney, December.

[Sutskever et al.2014] Y. Sutskever, O. Vinyals, and Q.V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

[Vaswani et al.2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

[Weaver1995] W. Weaver. 1995. Translation (1949). In *Machine Translation of Languages*. In: MIT Press, Cambridge, MA.

[Weiscbedel et al.] Ralph Weiscbedel, Jaime Carbonell, Barbara Grosz, Wendy Lehnert, Mitchell Marcus, Raymond Perrault, and Robert Wilensky. White Paper on Natural Language Processing.

[Williams et al.1988] R. J. Williams, G. E. Hinton, and D. E. Rumelhart. 1988. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, October.

[Wolk and Marasek2014] K. Wolk and K. Marasek. 2014. Polish - English Speech Statistical Machine Translation Systems for the IWSLT 2014. *CoRR*, abs/1509.09097.

[Wołk and Marasek2015] K. Wołk and K. Marasek. 2015. Neural-based Machine Translation for Medical Text Domain. Based on European Medicines Agency Leaflet Texts. *Procedia Computer Science*, 64:2 – 9.

[Wu et al.2016] Y. Wu, M. Schuster, Z. Chen, Q.V. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *CoRR*, abs/1609.08144.

[Zhou et al.2016] J. Zhou, Y. Cao, X. Wang, P. Li, and W. Xu. 2016. Deep Recurrent Models with Fast-Forward Connections for Neural Machine Translation. *TACL*, 4:371–383.